REPORT DOCUMENTATION PAGE

AFRL-SR-AR-TR-04-

0488

1. AGENCY USE ONLY (Leave blank)	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
	8/24/04	FINAL REPORT – 12/1/00 – 4/30/04

4. TITLE AND SUBTITLE
Increasing the Efficiency, Functionality, and Scope of Plan Management Agents

5. FUNDING NUMBERS
F49620-01-1-0066

6. AUTHOR(S)
Martha E. Pollack

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
The Regents of
The University of Michigan
3000 South State Street
Ann Arbor, Michigan 48109-1274

8. PERFORMING ORGANIZATION REPORT NUMBER
F003952 / 040994 Final Report

9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)
USAF, AFRL
AF Office of Scientific Research
4015 Wilson Blvd. Room 713
Arlington, VA 22203-9727

Dr. Robert L. Herklotz
Program Manager

10. SPONSORING / MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES

12a. DISTRIBUTION / AVAILABILITY STATEMENT
Approved for public release, distribution unlimited

Approved for public release, distribution unlimited

12b. DISTRIBUTION CODE

13. ABSTRACT *(Maximum 200 Words)*

As technology for automated plan generation has matured, there has been increased interest in techniques for managing plans, i.e., for tracking, updating, dispatching, and revising them in dynamic, uncertain environments. This project focused on extending computational techniques for plan management so as to

- increase the *efficiency* of processing so that large sets of plans can be managed;
- augment the *functionality* of the plan management systems to make them more useful; and
- broaden the *scope* of plan management algorithms so that they can be used in a wider set of domains.

A particular emphasis was on managing plans with richly expressive temporal constraints, and thus significant attention was given to developing efficient algorithms for temporal constraint reasoning.

20040930 018

14. SUBJECT TERMS
Plan Management, Temporal Reasoning

15. NUMBER OF PAGES

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT

Increasing the Efficiency, Function, and Scope of Plan Management Agents

AFOSR Contract Number F49620-01-1-0066

Project Summary

As technology for automated plan generation has matured, there has been increased interest in techniques for managing plans, i.e., for tracking, updating, dispatching, and revising them in dynamic, uncertain environments. This project focused on extending computational techniques for plan management so as to

- increase the *efficiency* of processing so that large sets of plans can be managed;
- augment the *functionality* of the plan management systems to make them more useful; and
- broaden the *scope* of plan management algorithms so that they can be used in a wider set of domains.

A particular emphasis was on managing plans with richly expressive temporal constraints, and thus significant attention was given to developing efficient algorithms for temporal constraint reasoning.

The principal accomplishments achieved under this contract include:

- Formulation of the problems of plan update and dispatch as one of solving temporal constraint problems in general, and DTPs in particular.
- Design of an algorithm for solving Disjunctive Temporal Problems (DTPs) that is two orders of magnitude faster than the previous state-of-the-art;
- Development of an algorithm for least-commitment (i.e., maximally flexible) dispatch of plans encoded as DTPs;
- Construction of a new formalism, Conditional Temporal Networks (CTNs) for modeling *conditional* plans with richly expressive temporal constraints, and development of algorithms for reasoning with this formalism;
- Discovery of a problem with prior approaches to conditional planning that renders those approaches incomplete, and proof that this problem is corrected in the CTN framework;
- Proposal of approaches to finding optimal solutions to temporal networks with uncontrollable events (STP-Us) even when one cannot guarantee the successful execution of the plans represented by those networks.
- Development of improved techniques for monitoring the execution of plans with rich temporal constraints under conditions of uncertainty, and for interacting appropriately with a human who may be performing plan execution.
- Integration of the various algorithms for plan-management tasks in prototype unified plan-management assistant systems.

Project Results

We briefly survey the technical results of this project. More details can be found in the publications listed in the next section; these publications also contain citations to the related literature, which we do not repeat here.

Plan Management as Temporal Constraint Solving

A fundamental decision in our project was to model certain central components of plan management—specifically, plan tracking, updating, and dispatching—as problems of temporal constraint solving. The basic idea is to translate a plan specified in a traditional planning formalism into a set of temporal constraints representing both temporal and causal structure. Plan update can then be viewed as a process of checking the consistency of the union of the constraints representing the initial plan, the constraints representing the new plan, and a set of constraints representing possible ways of resolving conflicts between the two plans; while plan retraction involves simply deletion from the constraint set. Similarly, plan execution can be modeled by adding to the set of constraints a new constraint that represents the execution time of an activity, and performing propagation to identify impact on the underlying plan. Failure to execute an action can be handled similarly (adding a non-execution constraint and propagating).

The cost of performing temporal reasoning depends on the type of constraints allowed. Fig. 1 gives our analysis of the space of temporal constraint formalisms. There are two classifying attributes: whether or not the constraints are allowed to include disjunctions, and whether or not they include conditional tags on actions and/or uncontrollable events, i.e., events whose time of execution is not the control of the execution agent. Some of the simpler formalisms were developed and investigated prior to the current project, including sets of constraints restricted to simple precedence relations, with or without disjuncts (labeled "Partial Order" and "Yang" in the diagram, respectively), Simple Temporal Networks (STPs), and Simple Temporal Networks with Uncertainty (STP-u). Others, while developed outside of the scope of the current project, were a focus of study in the current project, wherein we developed improved algorithms for reasoning with them; these include Disjunctive Temporal Problems (DTPs) and Probabilistic Simple Temporal Problems (PSTPs). Finally, the formalisms that allow conditional tagging, CSTPs and CDTPs, were developed within the scope of the current project.

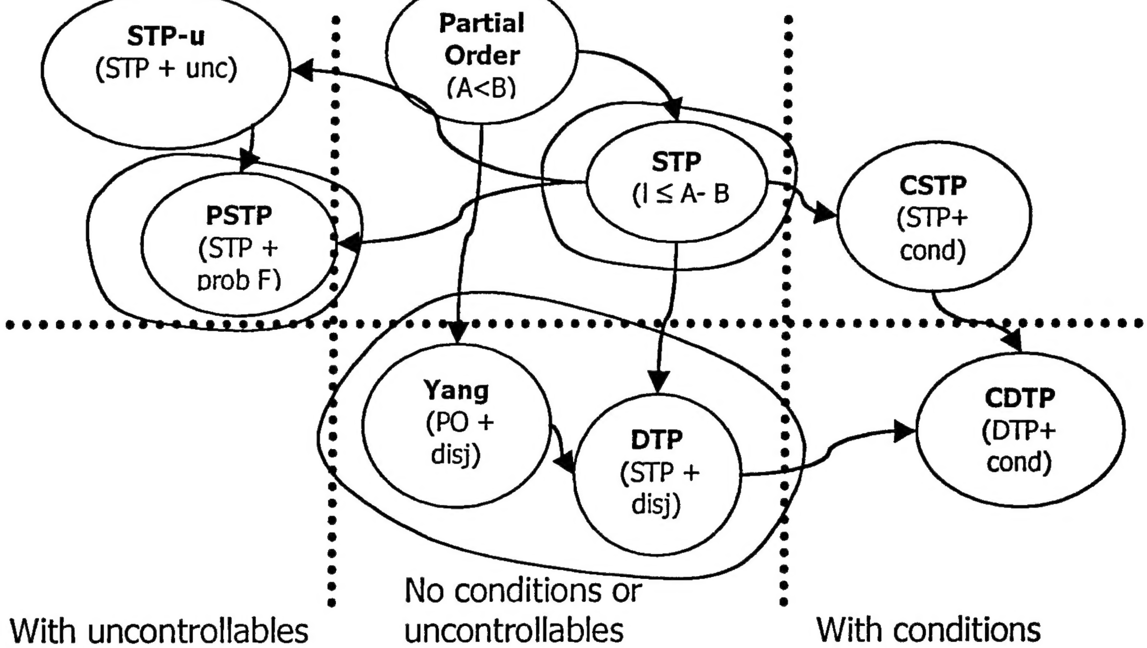

Figure 1. Space of Temporal Constraint Formalisms

Efficient DTP Solving

In the context of plan management, inclusion of disjunctive constraints is extremely important, because it provides a way of modeling actions that cannot overlap, and of modeling actions that present causal threats and thus must be either temporally demoted to precede the producer of the threatened constraint or promoted to follow the consumer. Unfortunately, solving Disjunctive Temporal Problems is NP-hard, in contrast to solving Simple Temporal Problems, which can be done in polynomial time using shortest-path algorithms. We there studied the DTP formalism, and developed efficient heuristic approaches to solving problems cast as DTPs. Specifically, we designed and implemented a DTP-solving system, Epilitis, and used it to conduct a thorough investigation of techniques for speeding up processing, including techniques that had been explored in previous work on DTPs. Integration of these techniques was a challenging technical problem. We also added no-good learning, a pruning strategy that had not previously been used in DTP-solving. Experimentation showed that no-good learning was a particularly powerful tool for speed-up in plan-merging problems. As a result of this work, we were able to achieve a two order-of-magnitude speed-up with Epilitis as compared to the previous state-of-the-art DTP solver. An example of our results is shown in Figure 2. In this graph, the Y-axis represents CPU time in seconds and the X-axis represents the ratio of constraints to variables (denoted by N): this metric has been established in the previous literature as defining the critical region of a DTP. All CSPs have a critical region, which is where speed-up has the greatest effect because base time to solve increases exponentially there. As can be seen, Epilitis performs particularly well in this region.

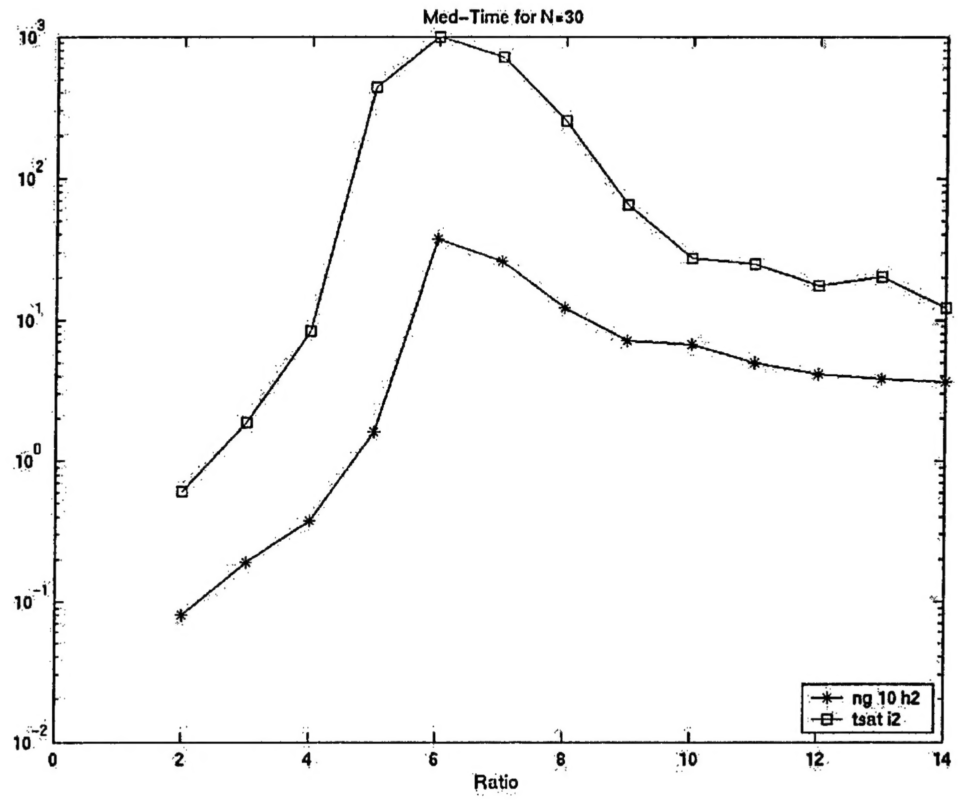

Figure 2. Performance of Epilitis compared to TSAT algorithm. Note that the y-axis is logarithmic.

Least-Commitment DTP Dispatch

In addition to maintaining a consistent model of a set of plans, it is important for a plan-management system to be able to dispatch those plans, i.e., to decide when to execute each action in the plan so as to ensure that all the constraints are satisfied. The difficulty of dispatching a plan depends on the nature of the plan and the environment in which it is to be executed. The simplest case arises when (1) a plan includes a specific time for the performance of each of its actions, and (2) it is to be executed in a static setting, one in which the only changes are the direct result of the plan execution itself. In this circumstance, plan dispatch is trivial: all that is required is for each action to be performed at its specified time. But most real-world planning and execution applications are not so simple. The evolution of the world is generally not fully known in advance, and thus it is difficult to give precise specifications of the times and durations of actions. Allowing for flexible constraints can make it possible to accommodate unanticipated events, but also makes dispatch more complicated, because there is no longer a unique point in time at which each action is to be performed. STPs and DTPs were developed to represent flexible quantitative temporal constraints; of the two, DTPs are more complicated, allowing disjunctions, and flexible dispatch of plans encoded as DTPs is similarly more challenging.

We developed an algorithm for flexibly dispatching plans encoded as DTPs, and showed that the algorithm satisfied four properties. (1) It is correct, i.e., whenever the executive executes actions according to the dispatch notifications, the performance of those actions respects the temporal constraints of the underlying plan. (2) It is deadlock-free: it provides enough information so that the executive does not violate a constraint through inaction. (3) It is maximally flexible, i.e., it does not issue a notification that unnecessarily eliminates a possible execution, i.e., an execution that respects the constraints of the underlying plan. (4) It is useful, i.e., it produces outputs that require only polynomial-time reasoning on the part of the executive. These properties, which we identified and formalized, must be at least partially satisfied by any dispatch algorithm to be valuable in a plan-management system.

The algorithm works by maintaining two pieces of information: an Execution Table, which specifies what actions may be performed, and a Deadline Formula, which specifies what actions must be performed by whatever deadline is nearest. The latter is computed by finding a minimal set cover for the set of consistent component STPs for the current DTP (i.e., the original DTP with all execution constraints up to the current time added). This fact suggests a way of providing the algorithm with anytime properties, at the cost of losing some flexibility: instead of computing the complete set of consistent component STPs after each change in plan status, one can limit the number computed based on available resources, and then finding the Deadline Formula for the (reduced) set.

Conditional Temporal Networks

The temporal reasoning formalisms that had been previously developed did not support the modeling of conditional plans, i.e., plans that are executed under conditions of uncertainty, where decisions about what actions to perform depend upon the outcome of contingent events that cannot be predicted with certainty at plan time. To enable plan-management systems to handle such plans, we developed the Conditional Temporal Problem (CTP) formalism, an extension of the temporal constraint-satisfaction processing models used in non-conditional temporal planning. Specifically, we augmented the previous models by (1) adding observation nodes, which correspond to the time-points at which observation actions end, and (2) attaching labels to all other nodes in the network. A node's label indicates the situation(s) in which the event it denotes will be executed. Observation actions are used to determine the value of labeled events.

The CTN framework allows for the off-line construction of conditional plans that are guaranteed to satisfy complex temporal constraints. Importantly, this can be achieved even while allowing for the decisions about the precise timing of actions to be postponed until execution time, in a least-commitment manner, thereby adding flexibility and making it possible to adapt the plan dynamically, during execution, in response to the observations made. A key result was the identification of three types of consistency in CTNs, which parallel consistency modeling in STP-U's: strong consistency, in which there exists a single solution that is valid regardless of the outcome of the contingent events; weak consistency, in which there exists a (possible separate) solution for each

possible set of outcomes; and dynamic consistency, in which, while there may not be a single always-valid solution, it is always possible to construct a solution dynamically, during the process of execution. That is, if a network is dynamically consistent, then information about the outcome of contingent events will always become available "early enough" to allow the plan management system to determine when to schedule any dependent events. Dynamic consistency is the most important property for real-world plan-management systems, and we showed that dynamic consistency can be determined by solving a disjunctive temporal problem constructed from the original CTP by adding a set of disjunctive constraints that model the contingencies. Thus, we can apply our results on efficient DTP solving to the problem of checking the consistency of CTPs.

We also showed that, even for plans without explicit quantitative temporal constraints, our approach fixes a problem that we discovered in earlier approaches to conditional planning, one that resulted in their being incomplete.

STP-Us that are not Dynamically Consistent

Where the work on CTPs was directed at reasoning with plans with actions that depend on continent events, STP-Us were developed by other researchers to model plans with events whose precise timing cannot be controlled, and which are thus called uncontrollables. A very attractive feature of STP-u's is that one can determine in polynomial time whether a given STP-u is dynamically controllable, i.e., whether there is a guaranteed means of execution such that all the constraints are respected, regardless of the exact timing of the uncertain events. (Note the parallelism between this notion and the notion of dynamic consistency in CTPs.) Unfortunately, if an STP-u is not dynamically controllable, limitations of the formalism prevent further reasoning about the probability of legal execution. In fact, because the STP-u formalism models only an interval during which an uncontrollable may occur and not a probability distribution over that interval, it is impossible for it to support the type of reasoning needed when dynamic controllability fails. We therefore adopted a related formalism that does include explicit distributions, the Probabilistic Simple Temporal Problems (PSTPs), and we showed that while it is difficult to compute the exact probability of legal execution, there are methods for bounding the probability both from above and below. We sketched alternative candidate algorithms of this purpose; work on their development is continuing in our follow-on AFOSR contract. Note that computing the probability of legal execution allows a temporal planner to decide, when uncertainty is present, whether to accept or reject candidate plans. In addition, lower bound computation has an important side-effect: it provides guidance as to how to execute an STP-u even when it is not dynamically controllable.

Execution Monitoring and Effective Reminding

While the central tasks for plan management consist of tracking, updating, and dispatching plans, there are other tasks that must be performed as well. These include monitoring plan execution, and, in the case in which the executive is a human,

deciding whether and when to interact with him. We therefore also addressed these tasks in this project.

The execution-monitoring problem is a difficult one when you allow the type of flexibility in plan representation that we have: rich temporal constraints and conditional branches. Because execution monitoring necessarily involves probabilistic reasoning to relate observed actions to individual or sets of planned activities, a Bayesian net formalism is the approach of choice. There have been two main prior approaches to using Bayesian nets in dynamic settings: time nets and their extensions to Dynamic Object Oriented Bayes net . However, each of these approaches has a limitation for execution monitoring. Time Nets can model quantitative temporal constraints, but cannot easily model the evolution of a belief over time. In contrast, the dynamic Bayes net formalisms, including DBNs and DOOBNs, are designed specifically to model belief evolution, but since they make the Markov assumption, they give up the capability to model arbitrary quantitative temporal constraints. We therefore developed an alternative, hybrid approach. It consists of three parts: a dynamic model that represents the stochastic evolution of beliefs over time, a model that reasons about the quantitative temporal relationship between events, and a mechanism that manages the flow of temporal information between the two. The approach was fully implemented and used in our prototype plan-management system, where it generally performed well. However, due to difficulties in formally proving its correctness, we are now exploring additional alternatives.

Execution modeling produces a picture of what the user has already done and what he is likely to do in the near future. This information must be combined with information about what the user is committed to, in order to make principled decisions about what reminders to issue and when. We developed techniques for making just such decisions. To produce an appropriate reminder, it is necessary to:

- identify which activities may require reminders based on their importance and their likelihood of being forgotten, which can be determined from the user model constructed as a result of execution monitoring,
- determine the most effective times to issue each reminder, taking account of the expected user behavior, as encoded in the user model, and any explicitly encoded preferences,
- balance reminder stability with dynamic rescheduling when changes to the plan occur; and,
- where possible and desirable, provide justifications as to why particular activities warrant a reminder.

It is relatively easy to create one reminder plan: one can simply issue a reminder at the earliest possible start time of every activity. However, such a reminder plan is not likely to be high-quality. Producing a high-quality reminder plan is more difficult, as user preferences, interactions among activities, and user actions taken so far should be considered. We thus adopted a local search/iterative refinement approach, in which we create the initial reminder plan as just suggested (reminders at the earliest possible time), and then perform local search, using a set of rewrite rules to generate

alternative candidate reminding plans. Again, this approach has been fully implemented and integrated into prototype systems.

Publications supported by the Project

1. M. Beetz, M. Ghallab, J. Hertzberg, and M. E. Pollack, editors, *Plan-Based Control of Robotic Agents,* LNCS/LNAI #2466, Springer-Verlag, 2002.

2. A. Berfield, P. Chrysanthis, I. Tsamardinos, M. E. Pollack, and S. Banerjee, "A Scheme for Integrating e-Services in Establishing Virtual Enterprises," *12th International Workshop on Research Issuess in Data Engineering,* 2002.

3. D. Colbry, B. Peintner, and M. E. Pollack, "Execution Monitoring with Quantitative Temporal Bayesian Networks," *6th International Conference on AI Planning and Scheduling,* Apr., 2002.

4. P. Ganchev, "Flexibility Measures of Sets of Plans," University of Pittsburgh Intelligent Systems Program M.S. Project Report, June, 2001.

5. J. F. Horty and M. E. Pollack, "Evaluating New Options in the Context of Existing Plans," *Artificial Intelligence,* 127(2):199-220, 2001.

6. B. Peintner and M. E. Pollack, "Low-Cost Addition of Preferences to DTPs and TCSPs," *19th National Conference on Artificial Intelligence,* July, 2004.

7. M. E. Pollack, L. Brown, D. Colbry, C. E. McCarthy, C. Orosz, B. Peintner, S. Ramakrishnan, and I. Tsamardinos, "Autominder: An Intelligent Cognitive OrthoticSystem for People with Memory Impairment," *Robotics and Autonomous Systems,* 44(3-4):273-282, 2003.

8. M. E. Pollack, C. E. McCarthy, S. Ramakrishnan, and I. Tsamardinos, "Execution-Time Plan Management for a Cognitive Orthotic System," in M. Beetz et al., editors, *Plan-Based Control of Robotic Agents,* 2002.

9. M. E. Pollack and I. Tsamardinos, "Efficiently Dispatching Plans Encoded as Simple Temporal Problems," to appear in I. Vlahavas and D. Vrakas, eds., *Intelligent Techniques for Planning,* 2004.

10. P. Schwartz and M. E. Pollack, "Planning with Disjunctive Temporal Constraints," *ICAPS-04 Workshop on Integrating Planning into Scheduling,* June, 2004.

11. I. Tsamardinos, "Constraint-Based Temporal Reasoning Algorithms, with Applications to Planning," University of Pittsburgh Intelligent Systems Program Ph.D. dissertation, Aug. 2001.

12. I. Tsamardinos and M. E. Pollack, "Efficient Solution Techniques for Disjunctive Temporal Reasoning Problems," *Artificial Intelligence* 151(1-2):43-90, 2003.

13. I. Tsamardinos, M. E. Pollack, and P. Ganchev, "Flexible Dispatch of Disjunctive Plans," *6th European Conference on Planning,* Oct. 2001.

14. I. Tsamardinos, M. E. Pollack, and S. Ramakrishnan, "Assessing the Probability of Legal Execution of Plans with Temporal Uncertainty, Workshop on Planning under Uncertainty and Incomplete Information, 13th International Conference on Planning and Scheduling, June, 2003.

15. I. Tsamardinos, T. Vidal, and M. E. Pollack, "CTP: A New Constraint-Based Formalism for Conditional, Temporal Planning," *Constraints: An International Journal,* Kluwer Academic Publishers, 8(4):365-383, 2003.

Personnel supported by the Project

This project provided support for the Principal Investigator, Dr. Martha Pollack, as well as several graduate students: Cheryl Orosz, Sailesh Ramakrishnan, Peter Schwartz, and Joseph Taylor.

Transitions

A number of the algorithms we developed, along with the prototype integrated plan-management system implemented in the current project, are now being extended and used in the DARPA CALO effort being undertaken at SRI International, on which we have a subcontract. Additionally, we are using this system in other research to develop assistive technology for people with cognitive impairment, with funding from the National Science Foundation.